In presenting the dissertation as a partial fulfillment of the requirements for an advanced degree from the Georgia Institute of Technology, I agree that the Library of the Institute shall make it available for inspection and circulation in accordance with its regulations governing materials of this type. I agree that permission to copy from, or to publish from, this dissertation may be granted by the professor under whose direction it was written, or, in his absence, by the Dean of the Graduate Division when such copying or publication is solely for scholarly purposes and does not involve potential financial gain. It is understood that any copying from, or publication of, this dissertation which involves potential financial gain will not be allowed without written permission.

7/25/68

AN ALGORITHM FOR MAXIMAL FLOW WITH GAINS

IN A SPECIAL NETWORK

A THESIS

Presented to

The Faculty of the Division of Graduate

Studies and Research

by

Anthony Michael Jezior

In Partial Fulfillment

of the Requirements for the Degree

Master of Science

in the School of Industrial and Systems Engineering

Georgia Institute of Technology

December, 1970

AN ALGORITHM FOR MAXIMAL FLOW WITH GAINS

IN A SPECIAL NETWORK

Approved:

Chairman

Date approved by Chairman 12/15/70

ACKNOWLEDGMENTS

My sincerest appreciation is extended to the members of my thesis reading committee for their assistance and helpful comments. A special debt of gratitude is extended to my faculty advisor, Dr. J. J. Jarvis for his patience, encouragement, assistance and advice.

My wife Bonnie receives my gratitude for her encouragement and the gentle prodding which assisted me in concluding this thesis.

Finally, I wish to acknowledge the excellent job Mrs. Claudine Taylor did in transcribing my handwriting into legibility.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF ILLUSTRATIONS

CHAPTER I

INTRODUCTION

1. Preliminary Concepts

Network flow theory relates mathematical graph theory to optimization mathematics of operations research. Extensive discussion in this field is available in the literature.

Berge (3) defines a graph as:

a. A set X

b. A function Γ mapping X into X.

The set X are called nodes and the function Γ a binary relationship which is represented by the arcs connecting the nodes. This definition could be pictorially described by a collection of points in a plane, representing nodes, and the lines connecting them, representing the arcs. Figure 1 shows a graph with $X = \{x, y, z\}$ and the binary relationship represented by the arcs (x, y), (y, z), (x, z), (z, x) and (y, y). The arcs in Figure 1 have a specific orientation and are defined as directed arcs. The graph is called a directed graph.

A transportation network evolves from a directed graph that is connected, contains no loops and in which the following conditions are satisfied:

a. There is a set of nodes which constitutes a source set and is denoted by S.

b. There is a set of nodes which constitutes a sink set and is denoted by T.

c. There is a nonnegative number associated with each arc defined as the arc

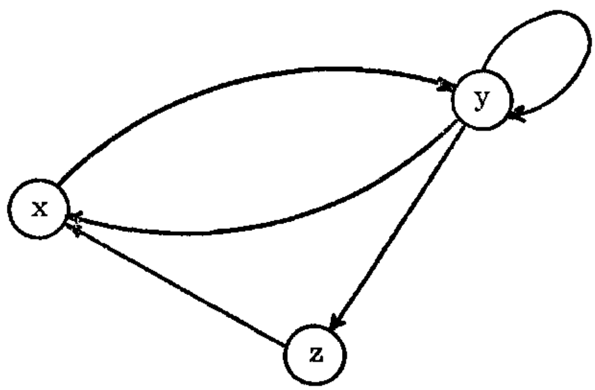

Figure 1. Directed Graph.

capacity and denoted by c(x, y). The value c(x, y) can be thought of as the maximum amount of some commodity which could be transported by the arc.

The sets S and T are mutually exclusive. Using a technique described in Ford and Fulkerson (8) these two sets can be modified so that the source and sink each comprise a single node. Figure 2 shows a transport network with the numbers on the arcs describing arc capacities.

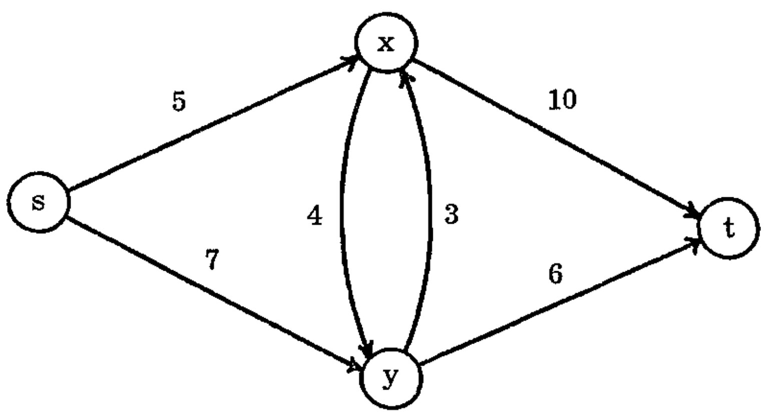

Figure 2. Transport Network.

A flow in the transportation network is the assignment of a nonnegative number, $f(x, y)$ to each arc (x, y) such that the following conditions are satisfied:

$$\text{(a)} \quad \sum_y f(x,y) - \sum_y f(y,x) = \begin{cases} v & x = s \\ 0 & x \neq s, t \\ -v & x = t \end{cases}$$

$$\text{(b)} \quad f(x,y) \leq c(x,y) \forall (x,y)$$

The equations (a) are called the node conservation equations; the inequalities (b) are the arc capacity constraints.

Determining the value of the maximal flow in a network can be expressed as a linear programming problem:

$$\text{(a)} \quad \text{Maximize } v \tag{1.1}$$

Subject to:

$$\text{(b)} \quad \sum_y \left[f(x,y) - f(y,x) \right] = \begin{cases} v & x = s \\ 0 & x \neq s, t \\ -v & x = t \end{cases}$$

$$\text{(c)} \quad 0 \leq f(x,y) \leq c(x,y)$$

Extensive discussion of solution techniques for network flows can be found in Ford and Fulkerson (8).

A network with gains is a network with n nodes and m arcs with $k(x, y)$, a multiplier, on arcs (x, y). The value $k(x, y)$ multiplies the flow out of node x on arc (x, y) to produce $k(x, y)\, f(x, y)$ units of flow at y.

The mathematical definition for a maximal flow with gains problem is

$$\text{(a) Maximize } v_s \tag{1.2}$$

$$\text{Subject to: (b) } \sum_y [f(x,y) - k(y,x)\, f(y,x)] = \begin{cases} v_s & x = s \\ 0 & x \neq s,t \\ -v_t & x = t \end{cases}$$

$$\text{(c) } 0 \le f(x,y) \le c(x,y)$$

When all $k(x,y) = 1$ formulation 1.2 reverts to 1.1. Hereafter we shall refer to problem 1.1 as the "unit gains problem."

Three additional terms in network theory which will be utilized in this thesis are defined here for completeness. A tree is a connected graph with no cycles. A forest is a graph consisting of one or more unconnected trees. An acyclic network is a network with no directed cycles. An acyclic network is always a directed network, but a directed network is not necessarily an acyclic network. We now present several applications of flow with gains in an acyclic network.

2. Background

A Rand Corporation study (17) and a paper by Ralph T. Murray (15) noted that delivery of health care, especially to the poor; effective cost techniques, and quantitative analysis of alternative increments to a city's already existing medical service capabilities are fruitful research areas. In a presentation (9) network flow theory was shown to be a promising optimization tool for budgetary decision making in health care. The problem of health care was viewed as a revised version of Murray's (15) proposals. Figure 3 depicts a schematic diagram of a possible

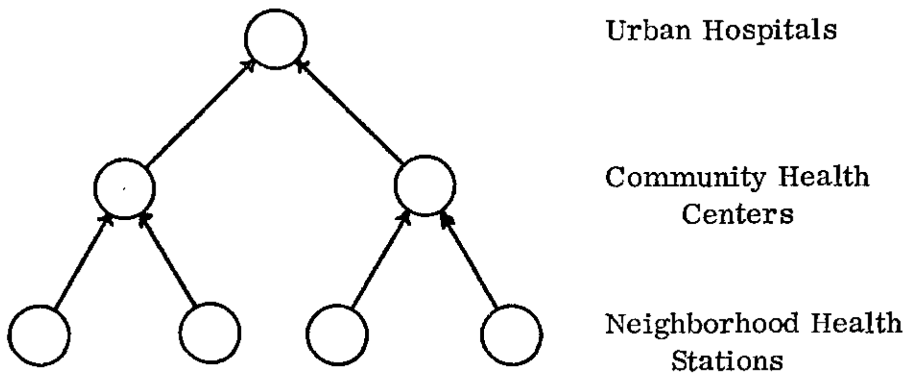

Figure 3. Health Care Systems.

system built toward a city's existing capabilities.

Taking into account differing aspects of care provided at each level in the system it is possible to formulate the health system as a transportation network with gains. Considering the facilities at each level as the nodes, it can be seen that for patients flowing through the system not all would pass from source to sink, i.e. not all patients going to community health centers pass to hospitals. Therefore, patients could be termed lost to the system. This loss factor may be expressed as a multiplier, $k(x, y)$. Thus the health care system can be viewed as a network with gains. In this case $k(x, y)$ would be restricted, that is $0 < k(x, y) \leq 1$. The network has the following special features:

a. All $k(x, y) > 0$.

b. All arcs are directed.

c. The graph is acyclic.

There are other potential applications of this specially structured network. A multi-level maintenance system where not all equipment passes through all levels provides a potential application. Educational or training systems and military personnel systems provide other potential areas.

In these applications the initial goal would be to determine the maximum flow into the system consistent with system capacity.

3. Objective

The objective of this research is to develop an efficient algorithm to determine the maximal flow out of the source of an acyclic network with gains having positive multipliers. An extension to this algorithm is also developed which maximizes flow into the sink. The maximal flow algorithm developed provides the first step for further research into an efficient algorithm for optimal flow with gains considering costs.

CHAPTER II

LITERATURE SEARCH

The network with gains problem was defined at 1.2. The formulation at 1.2 maximizes flow out of the source, defined as v_s. Flow into t is defined as v_t.

Differences to be noted between 1.1 and 1.2 are:

a. In 1.1, $v_s = v_t$; in 1.2 v_s may not necessarily be equal to v_t due to the effect of the multipliers.

b. A node-arc incidence matrix of 1.1 consists of $\pm$ 1 as elements. The node-arc incidence matrix of 1.2 depends upon k(x, y). With $\pm$ 1 as matrix elements the solution for v in 1.1 will always be integral while 1.2 may not necessarily be integral for v_s.

c. The presence of multipliers complicates network solution procedures, especially because of $v_s \neq v_t$.

The literature for maximal flow with gains is sparse. The most definitive work has been done by Jewell (10). He investigated the problem of optimal flow with gains through a general network. The formulation of Jewell's problem is:

$$\text{(a)} \quad \text{Maximize} \sum_{x,y} a(x,y)f(x,y) \tag{2.2}$$

Subject to:

$$\text{(b)} \quad \sum_{y}\left[f(x,y) - k(y,x)f(y,x)\right] = \begin{cases} Q & x = s \\ 0 & x \neq s \end{cases}$$

(c) $0 \leq f(x, y) \leq c(x, y)$

Q defines some total input flow requirement and is not a variable in this formulation. The cost per unit of flow on (x, y) is defined as a (x, y). The $k(x, y)$ in Jewell's formulation can assume any value other than zero. An arc is constructed from s to t with $k(x, y) = -1$ which accounts for absence of v_t in 2.2. This boundary arc has no effect on Q, but balances $f(x, y)$ at t.

Jewell uses a primal-dual technique in his solution procedure which includes a maximal flow subroutine. This subroutine detects network structures which can absorb flow into the network. Comparing Jewell's subroutine to the one developed in this paper, the following points are evident:

a. Jewell's technique solves the problem for the general case with $k(x, y)$ having any non-zero value. The algorithm developed here applies to the specially structured network described in Chapter I.

b. Jewell's subroutine is essentially a primal technique, i.e. he needs make no assignment of dual variables until the end and, throughout, works with the entire original network. By contrast, the algorithm of this thesis utilizes a primal-dual technique--constantly changing dual variables and working with subnetworks based on the dual variables.

c. Both algorithms use variations of the Ford-Fulkerson labelling procedure. However, Jewell's application is a complicated labelling process. In Jewell's labelling process labelled nodes may be relabelled many times. Each time relabelling occurs it is necessary to determine whether one of the labels was derived from the other, requiring a tracing back through the network. In the algorithm presented here nodes are labelled once and only once during any labelling process.

d. Flow changes are carried along in our labels; when flow is assigned it is easily done on one pass from t to s. When flow is assigned in Jewell's algorithm a complicated procedure is used which requires two labellings back to the source for each node involved.

e. Because of the specially structured network and procedures adopted which are proposed by Johnson (11, 12), the flow with gains into a special network presented here is finite where Jewell's may not be as originally presented. Under Johnson's modification, Jewell's algorithm would be finite but would require a triple labelling procedure.

Berge and Ghouila-Houri (9) discuss aspects of the flow with gains problem. Their problem is defined with multipliers on nodes. To obtain the formulation with multipliers on arcs, a vertex is constructed in the middle of the arc with associated multipliers. Two problems are shown in (9):

a. Minimization of total time of several machines, a problem attributed to Iri, Amari and Takata.

b. Minimization of transportation cost of fuel supply to power stations.

Charnes and Raike (5) discuss a one pass algorithm for acyclic networks with prices on arcs. At each node a quantity representing supply or demand is given. The problem maximizes total revenue while satisfying supply and demand restrictions.

Two algorithms which solve the above problem are presented in (5) which in both cases disregard capacity constraints. The procedure used in (5) to establish the initial dual variables will be modified for use in this paper. This procedure is as follows:

a. Assign node s, $\pi(s) = 0$.

b. Select node y such that $\pi(x)$ have been assigned node numbers for all x which are incident with arcs directed into node y.

c. Determine $\pi(y)$ by $\pi(y) = \max_{x} \{\pi(x) + a(x, y)/k(x, y)\}$.

d. When $\pi(t)$ is assigned flow is assigned along a path from f which $\pi(x)$ assigned.

The techniques described in (5) are especially applicable to PERT networks and may also provide an initial dual-feasible solution to the capacitated problem.

Other variations of the flow with gains problem have been examined. Balas (1), Balas and Ivanescue (2), and Eisemann (7) use transportation tableaus in solution techniques for bipartite networks. Dantzig (6) and Markowitz (14) make use of the near-triangularity of the basis in solution techniques.

Johnson (11, 12) notes that in the flow with gains problem solutions must be kept basic or algorithms might never converge in a finite number of steps or even to the optimal solution. Johnson also shows that for non-integer capacities the Ford-Fulkerson algorithm for the unit gains problem would not converge in a finite number of steps. We shall make use of Johnson's findings and his proposed degeneracy modification for the Ford-Fulkerson algorithm in ensuring that the algorithm for flow with gains in the specially structured network converges.

CHAPTER III

THE MAXIMAL FLOW WITH GAINS ALGORITHM

1. General Considerations

In this chapter an algorithm for maximizing the flow into a network with gains is developed and justified. An extension will be shown where flow can be maximized into the sink. As described earlier, the network (with n nodes and m arcs) will contain only directed arcs, be acyclic and have all $k(x, y) > 0$.

The primal problem was formulated in 2.1 but will be repeated here for ease in developing and justifying the algorithm.

$$\text{(a)} \quad \text{Maximize } v_s \tag{3.1}$$

$$\text{(b)} \quad \sum_y \left[f(x, y) - k(y, x)f(y, x)\right] = \begin{cases} v_s & x = x \\ 0 & x \neq s, t \\ -v_t & x = t \end{cases}$$

$$\text{(c)} \quad 0 \leq f(x, y) \leq c(x, y)$$

$$\text{(d)} \quad v_t, v_s \geq 0$$

Designating $\pi(x)$ as the dual variables for 3.1b. and $\gamma(x, y)$ for 3.1c., the dual of 3.1 is:

$$\text{(a)} \quad \text{Minimize } \sum_{x,y} c(x, y)\, \gamma(x, y) \tag{3.2}$$

(b) Subject to: $\pi(s) \leq -1$

(c) $\pi(x) - k(x, y)\, \pi(y) + \gamma(x, y) \geq 0 \;\forall (x, y)$

(d) $\pi(t) \geq 0$

(e) $\gamma(x, y) \geq 0$

Applying Kuhn-Tucker conditions to (3.1) and by taking:

$$\gamma(x, y) = \max\{0, k(x, y)\, \pi(y) - \pi(x)\}$$

optimality criteria are determined to be:

$$\text{(a)} \quad \pi(x) - k(x, y)\, \pi(y) > 0 \Rightarrow f(x, y) = 0 \qquad (3.3)$$

(b) $\pi(x) - k(x, y)\, \pi(y) < 0 \Rightarrow f(x, y) = c(x, y)$

(c) $v_t > 0 \Rightarrow \pi(t) = 0$

(d) $v_s > 0 \Rightarrow \pi(s) = -1$

The criteria 3.3 a-d can also be determined using the Theorem of Complementary Slackness, a special case of Kuhn-Tucker related to linear programs. Conditions 3.3c, and d. are implied when $v_t > 0$ and $v_s > 0$, respectively. Throughout this paper 3.3 a-d will be referred to as complementary slackness conditions.

The algorithm will be initiated with a primal feasible solution and an "almost feasible" dual solution, i.e. one satisfying all dual constraints except 3.2d and all complementary slackness conditions except 3.3c. When $\pi(t) = 0$, an optimal solution is obtained.

Prior to initiating the algorithm it is necessary to guarantee a path from s to all other x in the network. The following procedure alluded to in (5) will accomplish this:

Step 1. Remove all arcs entering s and leaving t.

Step 2. Discard any node which has no arcs.

Step 3. Discard any node, except s, having only arcs leaving it and these arcs.

Step 4. Discard any node, except t, having only arcs entering it and these arcs.

2. Algorithm

This section will present the algorithm in its entirety. The next section will provide the justification.

Step 1 (Initialize Node Numbers)

a. Set $\pi(s) = -1$. All other $\pi(x)$ are not yet assigned.

b. Let $p(x) = \{y \mid y$ is an immediate predecessor of $x\}$, i.e. there exists an arc (y, x) in the network.

c. Select some x for which all $y \in p(x)$ have assigned node numbers.

$$\text{Set } \pi(x) = \min\left\{\frac{\pi(y)}{k(y, x)} \;\middle|\; y \in p(x)\right\}$$

d. Repeat step c until all nodes have assigned node numbers.

Step 2 (Flow Change)

a. Determine admissible arcs by the criteria:

$\pi(x) - k(x,y)\pi(y) = 0$ where $0 \le f(x,y) \le c(x,y)$

b. Label on the network of admissible arcs as follows:

(1) Assign s, $L(s) = [-, e(s) = \infty]$

(2) For any node y which is unlabelled, if node x is labelled and arc (x,y) is admissible with $f(x,y) < c(x,y)$, label node y with $L(y) = [x^+, e(y)]$ where $e(y) = k(x,y)\min[c(x,y) - f(x,y), e(x)]$. If node x is labelled and arc (y,x) is admissible with $f(x,y) > 0$ label node y with $L(y) = [x^-, e(y)]$ where $e(y) = \frac{1}{k(y,x)}$ $\min\{f(y,x), e(x)\}$.

c. When t is labelled, breakthrough is achieved and a flow augmenting path is determined. The flow is assigned as follows:

Denote the flow augmenting path, P, as $s = x_0, x_1 \ldots x_{k-1}, x_k = t$. Let $e'(t) = e(t)$. For $i = k - 1. \ldots 1$, $e'(x_i)$ is;

$$e'(x_i) = \begin{cases} \dfrac{e'(x_i + 1) \text{ if } L(x_{i+1}) = [x_i^+, e(x_{i+1})]}{k(x_i, x_{i+1})} \\ e'(x_{i+1})\, k(x_{i+1}, x_i) \text{ if } L(x_{i+1}) = [x_i^-, e(x_{i+1})] \end{cases} \tag{3.6}$$

Flow values are:

$$f'(x_i, x_{i+1}) = f(x_i, x_{i+1}) + e'(x_i) \text{ if } (x_i, x_{i+1}) \in P \tag{3.7}$$

$$f'(x_{i+1}, x_i) = f(x_{i+1}, x_i) - e'(x_{i+1}) \text{ if } (x_{i+1}, x_i) \in P$$

All other flows remain the same.

d. Erase labels; repeat step 1 with the new feasible flow, $f'(x,y)$, until non-breakthrough occurs.

Step 3 (Dual Variable Change)

When non-breakthrough occurs:

a. Let $X = \{x \mid x \text{ is labelled}\}$ and $\bar{X} = \{x \mid x \text{ is unlabelled}\}$. Determine sets of arcs A_1 and A_2 according to the criteria:

$$\text{(a)}\quad A_1 = \{(x,y) \mid x \epsilon X,\ y \epsilon \bar{X},\ \pi(x) - k(x,y)\,\pi(y) > 0\} \tag{3.8}$$

$$\text{(b)}\quad A_2 = \{(x,y) \mid x \epsilon \bar{X},\ y \epsilon X,\ \pi(x) - k(x,y)\,\pi(y) < 0\}$$

b. Determine θ where:

$$\text{(a)}\quad \theta = \max\{\theta_1, \theta_2, -1\} \tag{3.9}$$

$$\text{(b)}\quad \theta_1 = \max\left\{\frac{\pi(x) - k(x,y)\pi(y)}{k(x,y)\,\pi(y)} \;\middle|\; (x,y) \in A_1\right\}.$$

$$\text{(c)}\quad \theta_2 = \max\left\{-\frac{[\pi(x) - k(x,y)\pi(y)]}{\pi(x)} \;\middle|\; (x,y) \in A_2\right\}$$

$$(\theta_i = -\infty \text{ if } A_i = \Phi)\ .$$

If $\theta = -1$, the solution is optimal and the algorithm is terminated. Otherwise change $\pi(x)$ as follows:

$$\text{(a)}\quad \pi'(x) = \pi(x),\ x \in X \tag{3.10}$$

$$\text{(b)}\quad \pi'(x) = (1+\theta)\,\pi(x),\ x \in \bar{X}$$

With $\pi'(x)$ determined, repeat Step 2.

3. Justification of the Algorithm

The series of Lemmas, Corollaries and Theorems show that the algorithm achieves optimality in a finite number of steps while maintaining all primal constraints and complementary slackness conditions.

Lemma 1: Upon completion of the network reduction procedure, a path exists from s to every x and from every x to t.

Proof: We first prove by contradiction that a path exists from s to every x. Begin a labelling process at x. Choose any arc (y, x) entering x (the procedure guarantees at least one) and label node y. Repeat the labelling procedure from y until some node z is reached from which no labelling can be accomplished. No node in the sequence of labelled nodes can ever be repeated since the network is acyclic. Thus the labelling must terminate after at most $n - 1$ steps. If $z \neq s$, then since z has no arcs entering it we have a contradiction to the first step of the reduction procedure. Proof of the existence of a path from all x to t is by a similar procedure. Q.E.D.

Lemma 2: The modified shortest path algorithm terminates with $\pi(x) < 0, \forall x$ and $\pi(x) - k(x,y)\,\pi(y) \geq 0 \forall (x,y)$.

Modifying the technique of Charnes and Rake (5), that is, assigning $\pi(x)$ only if all immediate predecessors to x have $\pi(x)$ assigned is a central point in the proof. Charnes and Raike provide a proof, but a different proof is provided here which can be implemented without explicitly specifying P(x).

Proof: First it will be shown by construction that some node always exists where predecessors all have assigned values. Choose any x with $\pi(x)$ unassigned.

Check the predecessors of x, $y \epsilon p(x)$. If for some $y \epsilon p(x)$, $\pi(y)$ is unassigned, consider that y. Continue until some node z is reached with $\pi(y)$ assigned for all $y \epsilon p(z)$. That such a node z exists is apparent since the potential numbers of candidate nodes for each predecessor set is being reduced by one at each step and as indicated in Lemma 1, at worst, the process will terminate with $p(z) = \{s\}$ and s has assigned value.

Assign $\pi(z)$ the value:

$$\pi(z) = \min \left\{\frac{\pi(y)}{k(y, z)} \middle| y \epsilon p(z)\right\}$$

Iterate for some other node with unassigned node number until all $\pi(x)$ are assigned. Since one more node will have its node number assigned at each iteration, the modified shortest path algorithm terminates after n - 1 iterations.

To show $\pi(x) < 0$, consider any path, $P = (s = x_0, x_1, \ldots x_{j-1}, x_j = x)$.

When the modified shortest path algorithm terminates, $\pi(x_{i+1}) \leq \dfrac{\pi(x_i)}{k(x_i, x_{i+1})}$.

Substituting recursively for $\pi(x_i)$ we obtain $\pi(x) \leq \dfrac{\pi(s)}{\prod\limits_{i=0,\ldots j-1} k(x_i, x_{i+1})}$. But $\pi(s) = -1$ and $\prod\limits_{i=0,\ldots j-1} k(x, y) > 0$; therefore $\pi(x) < 0$. Since at any x the minimum operator guarantees that $\pi(y) \leq \dfrac{\pi(x)}{k(x, y)}$ thus we have $\pi(x) - k(x, y)\, \pi(y) \geq 0$ for $\forall$ (x, y).

Q.E.D.

Lemma 3: Primal constraints 3.1b. and c. are maintained throughout the algorithm.

Proof: Since we begin with all f(xy) = 0, the primal constraints are satisfied initially. We shall first consider 3.1b. Four cases shown in Figure 4 will be considered to show that the conservation equations are maintained upon completion of the flow augmenting routine.

Case 1 $\quad x_{i-1} \longrightarrow x_i \longrightarrow x_{i+1}$

Case 2 $\quad x_{i-1} \longleftarrow x_i \longleftarrow x_{i+1}$

Case 3 $\quad x_{i-1} \longrightarrow x_i \longleftarrow x_{i+1}$

Case 4 $\quad x_{i-1} \longleftarrow x_i \longrightarrow x_{i+1}$

Figure 4. Flow Changes.

Consider Case 3. From 3.7 we obtain:

$$e'(x_i) = k(x_{i+1}, x_i)\, e'(x_{i+1})$$

$$e'(x_{i-1}) = \frac{e'(x_i)}{k(x_{i-1}, x_i)}$$

At x_i the total flow out remains the same and the total flow in is changed by:

$-e'(x_i) + \dfrac{e'(x_i)k(x_{i-1}, x_i)}{k(x_{i-1}, x_i)} = 0.$ Thus the conservation equation holds with new flows for case 3 since they held previously. The proof for Cases 1, 2 and 4 follow in the same manner.

The capacity constraints 3.1c are continuously maintained throughout the algorithm by the labelling technique described in 3.5b. Choosing the minimum label at each node ensures that no arc capacity on the labelled path is violated. Q.E.D.

Step 3 of the algorithm provides the method for dual variable change. The following series of lemmas will show that changes made according to 3.9a-c in Step 3 will maintain the complementary slackness conditions on every arc while increasing $\pi(t)$ toward its optimal value zero.

Lemma 4: If at any iteration $\pi(x) < 0$, then $-1 \leq \theta < 0$.

Proof: Consider first the set of arcs A_1 where $\pi(x) - k(x,y)\,\pi(y) > 0$. The relationship to determine θ for arcs in A_1 is described at 3.9 b. Since $\pi(x) < 0, \forall x$ then $k(x,y)\,\pi(y) < 0$ and θ determined by 3.9b. results in $\theta < 0$. The same argument holds for θ determined by 3.9c for arcs in A_2 and results in $\theta < 0$.

Optimality criteria 3.3c requires that when $v_t > 0$, $\pi(t) = 0$. Any $\theta < -1$ will cause $\pi(t) > 0$ which violates 3.3d. Therefore θ takes on values, $\theta \geq -1$. Q.E.D.

Lemma 5: If $\pi(x) < 0$ and $-1 \leq \theta < 0$ then $\pi'(x) < 0$ or an optimal solution is obtained.

Proof: Consider $-1 \leq \theta < 0$. Dual variables are changed according to 3.10a, and b. If $\pi(x) < 0$, it is clear $\pi'(x) < 0$ for $-1 < \theta < 0$. For $\theta = -1$, $\pi'(x) = 0$,

$x \in \bar{X}$. Since $t \in \bar{X}$, $\pi'(t) = 0$ fulfilling feasibility condition 3.2d and optimality criterion 3.3c.

Corollary 1: At any iteration of the algorithm, either $\pi(x) < 0, \forall$ x or an optimal solution is obtained.

Proof: Lemma 2 guarantees that at the start of the first iteration of Step 2, $\pi(x) < 0$. Lemma 4 shows that if $\pi(x) < 0$ then $-1 \leq \theta < 0$. Lemma 5 ensures $\pi'(x) < 0$ or optimality is attained when $\pi(x) < 0$ and $-1 \leq \theta < 0$. Proceeding on an inductive basis the corollary is proven.

Q.E.D.

Choosing $\theta = \max\{\theta_1, \theta_2, -1\}$ as shown in 3.9a-d ensures that at least one more arc becomes admissible. It is now necessary to show that complementary slackness conditions 3.3a and b are maintained for any dual variable change.

Lemma 6: For the new dual variable at any iteration the complementary slackness conditions 3.3a and b will be maintained.

Proof: In X, no dual variable changes occur, therefore for (x, y) in the labelled set X, the conditions at 3.3a and b will be maintained.

Consider the sets A_1 and A_2 in $(X, \bar{X})$ and $(\bar{X}, X)$. The relations 3.9 b and c for θ were determined to satisfy conditions for these arcs.

Two other conditions at $(X, \bar{X})$ and $(\bar{X}, X)$ must be investigated. The first is $(x, y) \in (X, \bar{X})$ and $\pi(x) - k(x, y)\, \pi(y) \leq 0$ for which we must have $f(x, y) = c(x, y)$. After the dual variable change $\pi(x) - k(x, y)\pi(y) < 0$ since $[(1 + \theta)\, \pi(y)] > \pi(y)$. Thus these arcs maintain complementary slackness. The other situation occurs when $(x, y) \in (\bar{X}, X)$ and $\pi(x) - k(x, y)\, \pi(y) \geq 0$ which must be accompanied by

$f(x, y) = 0$. At the next iteration $(1 + \theta)\ \pi(x) - k(x, y)\ \pi(y) > 0$ since $(1 + \theta)\ \pi(x) > \pi(x)$. Therefore complementary slackness conditions for these arcs are satisfied.

Any $(x, y) \in (\bar{\bar{X}}, \bar{X})$ will maintain its complementary slackness condition since $[\pi'(x) - k(x, y)\ \pi'(y)] = (1 + \theta)[\pi(x) - k(x, y)\ \pi(y)]$ and $(1 + \theta) > 0$.

Q. E. D.

The following series of lemmas will indicate that we can maintain basic feasible solutions at every iteration and that after finite intervals the primal objective function strictly increases. Thus finite convergence for the flow with gains algorithm when the network is acyclic will be proven.

Dantzig (6) and Johnson (11, 12) show that in the unit gains problem, flows strictly between their bounds must form a forest in order to correspond to a basic feasible solution. Johnson (11, 12) shows that in the flow with gains problem if the variables between their bounds form a forest the corresponding solution is basic. However, some basic solutions may form undirected cycles from variables strictly between their upper and lower bounds.

Lemma 7: At any iteration of the flow with gains problem in an acyclic network a forest, and thus a basic feasible solution, can be constructed from variables strictly between their upper and lower bounds.

Proof: Assume that at some iteration in the algorithm a flow cycle was formed with $0 < f(x, y) < c(x, y)$. Consider any x_i and x_j, $x_i \neq x_j$ in this cycle and define two paths:

$$P_1 = (x_i = x_0, x_1 \ldots x_p = x_j) \text{ and}$$

$$P_2 = (x_i = y_0, y_1 \ldots y_z = x_j).$$

Let P_1^+ be the set of forward arcs in P_1; and P_1^- the set of reverse arcs. Define P_2^+ and P_2^- in a similar manner. Suppose flow is increased from x_i along P_1 and decreased from x_i along P_2 successively by some ϵ until some (x, y) becomes either saturated or flowless. This breaks the cycle and if repeated for all cycles constructs a forest with variables strictly between their upper and lower bounds.

We must now ensure that conservation of flow is maintained. Considering the nodes in P_1 and P_2 not equal to x_i or x_j, the ϵ changes maintain conservation of flow by the labelling process. It is clear that the ϵ increase from x_i along P_1 and ϵ decrease along P_2, maintain conservation of flow at x_i. The final case to consider is at x_j.

Since on both P_1 and P_2, $0 < f(x,y) < c(x,y)$, we have $\pi(x) - k(x,y)\pi(y) = 0$. Thus:

$$\pi(x_e) = k(x_e, x_{e+1})\, \pi(x_{e+1}), \quad (x_e, x_{e+1}) \in P_1^+$$

and

$$\pi(x_e) = \frac{\pi(x_{e+1})}{k(x_{e+1}, x_e)}, \quad (x_{e+1}, x_e) \in P_1^-$$

Upon recursive substitutions in these equations we obtain:

$$\text{(a)} \quad \pi(x_i) = \frac{\prod_{P_1^+} k(x,y)}{\prod_{P_1^-} k(x,y)}\, \pi(x_j)$$

After a similar operation on P_2 we obtain:

$$\text{(b)} \quad \pi(x_i) = \frac{\prod\limits_{P_2^+} k(x,y)}{\prod\limits_{P_2^-} k(x,y)} \; \pi(x_j)$$

Setting (a) = (b) and dividing by non-zero $\pi(x_j)$ we obtain:

$$\frac{\prod\limits_{P_1^+} k(x,y)}{\prod\limits_{P_1^-} k(x,y)} = \frac{\prod\limits_{P_2^+} k(x,y)}{\prod\limits_{P_2^-} k(x,y)}$$

The right and left sides of the equality are precisely the gain factors along their respective paths. Thus ϵ changes as described will result in conservation of flow at x_j.

Q.E.D.

Johnson (12) has proposed a degeneracy modification for the Ford and Fulkerson flow problem which consists of the following procedure:

a. Form the set E_1 of (x, y) with $0 < f(x,y) < c(x,y)$.

b. After each breakthrough, first label in E_1 as much as possible.

c. Label in $\bar{E}_1$, but check (x, y) in E_1 each time a new node is labelled.

This modification or the procedure described in Lemma 7 applied to the flow with gains problem in an acyclic network ensures that a basic feasible solution consisting of a forest is available.

Lemma 8: The objective function in the algorithm for maximal flow with gains in the acyclic network must increase at finite intervals.

Proof: Consider the network with gains. At the start of any iteration, s is always labelled. When non-breakthrough occurs and $\theta > -1$ the dual variables are changed such that all nodes in x will be labelled again and at least either one arc, $(x, y) \in (X, \bar{X})$ becomes admissible with $f(x, y) = 0$ or one arc, $(x, y) \in (\bar{X}, X)$ becomes admissible with $f(x, y) = c(x, y)$. Thus at least one more $x \in \bar{X}$ can be labelled, and breakthrough must occur in at most $n - 1$ dual variable changes (resulting in a flow increase in v_s).

Q.E.D.

Theorem 1: The algorithm for maximal flow with gains in an acyclic network achieves an optimal solution in a finite number of steps.

Proof: Lemma 7 ensures that basic feasible solutions consisting of forests always exist. Lemma 8 shows that v_s increases in at least $n - 1$ iterations between flow changes. Since there are a finite number of basic feasible solutions the algorithm terminates in a finite number of steps.

Q.E.D.

4. Maximizing Flow into t

Consider the goal of maximizing the flow into the sink, t. The problem is formulated:

(a) Maximize v_t (3.11)

Subject to:

(b) $$\sum_y \left[f(x,y) - k(y,x)\, f(y,x) \right] = \begin{cases} v_s & x = s \\ 0 & x \neq s, t \\ -v_t & x = t \end{cases}$$

(c) $0 \le f(x,y) \le c(x,y)$

(d) $v_t,\ v_s \ge 0$

The dual is similar to 3.2 except 3.2b and d are changed to:

3.2 (b′) $\pi(s) \le 0$ (3.12)

3.2 (d′) $\pi(t) \ge 1$

The complementary slackness condition 3.3a and b hold. Those at 3.3c and d become:

3.3(c′) $v_t > 0 \Rightarrow \pi(t) = 1$ (3.13)

3.3(d′) $v_s > 0 \Rightarrow \pi(s) = 0$

The algorithm proceeds as follows:

Step 1 (Initialize Node Numbers)

a. Set $\pi(t) - 1$. All other $\pi(x)$ are not yet assigned.

b. Let $s(x) = \{y \mid y \text{ is a successor of } x\}$.

c. Select some x for which all $y \epsilon s(x)$ have been assigned node numbers.

Set $\pi(x) = \max\{k(x,y)\pi(y) \mid y \epsilon s(x)\}$.

d. Repeat step c until all nodes have been assigned node numbers.

Step 2 (Flow Change)

This step is accomplished as for maximizing v_s.

Step 3 (Node Number Change)

a. At $(X, \bar{X})$ determine A_1 and A_2 according to 3.8 a and b.

b. Determine θ where:

$$\theta = \max\{\theta_1, \theta_2, -1\}$$

$$\text{(a)} \quad \theta_1 = \max\left\{\frac{k(x,y)\,\pi(y) - \pi(x)}{\pi(x)} \;\Big|\; (x,y) \epsilon A_1\right\} \tag{3.14}$$

$$\text{(b)} \quad \theta_2 = \max\left\{\frac{-[k(x,y)\,\pi(y) - \pi(x)]}{k(x,y)\,\pi(y)} \;\Big|\; (x,y) \epsilon A_2\right\}$$

$$(\theta_i = -\infty \text{ if } A_i = \Phi)$$

If $\theta = -1$, the solution is optimal and the algorithm terminated.

Otherwise change $\pi(x)$, $x \epsilon X$.

$$\text{(a)} \quad \pi'(x) = (1 + \theta)\,\pi(x), \quad x \epsilon X \tag{3.15}$$

$$\text{(b)} \quad \pi'(x) = \pi(x), \quad x \epsilon \bar{X}$$

With $\pi'(x)$ determined, repeat Step 2. Justification of this extension will not be shown. The proofs concerning its justification are similar to those for maximizing v_s.

5. Example

Consider the network shown in Figure 5.

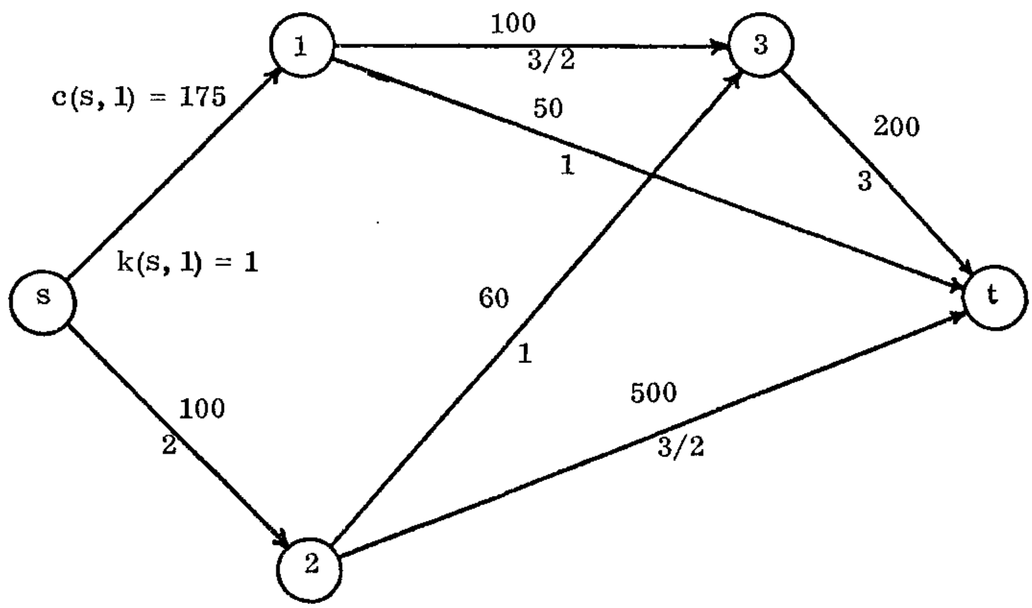

Figure 5. Example Network.

Step 1 (Initialize Node Numbers)

Setting $\pi(s) = -1$ and solving the modified shortest route problem values for $\pi(x)$ are as follows: $\pi(s) = -1$; $\pi(1) = -1$; $\pi(2) = -\frac{1}{2}$; $\pi(3) = \min\{-2/3, -1/2\} = -2/3$; $\pi(t) = \min\{-1, -2/9, -2/3\} = -1$.

Step 2 (Flow Change)

Figure 6 shows the network with dual variables and first iteration of labelling. Breakthrough occurs along the path: s, 1, t. Applying the flow augmenting routine, flow values along the path are: $f(s, 1) = 50$; $f(1, 5) = 50$. The

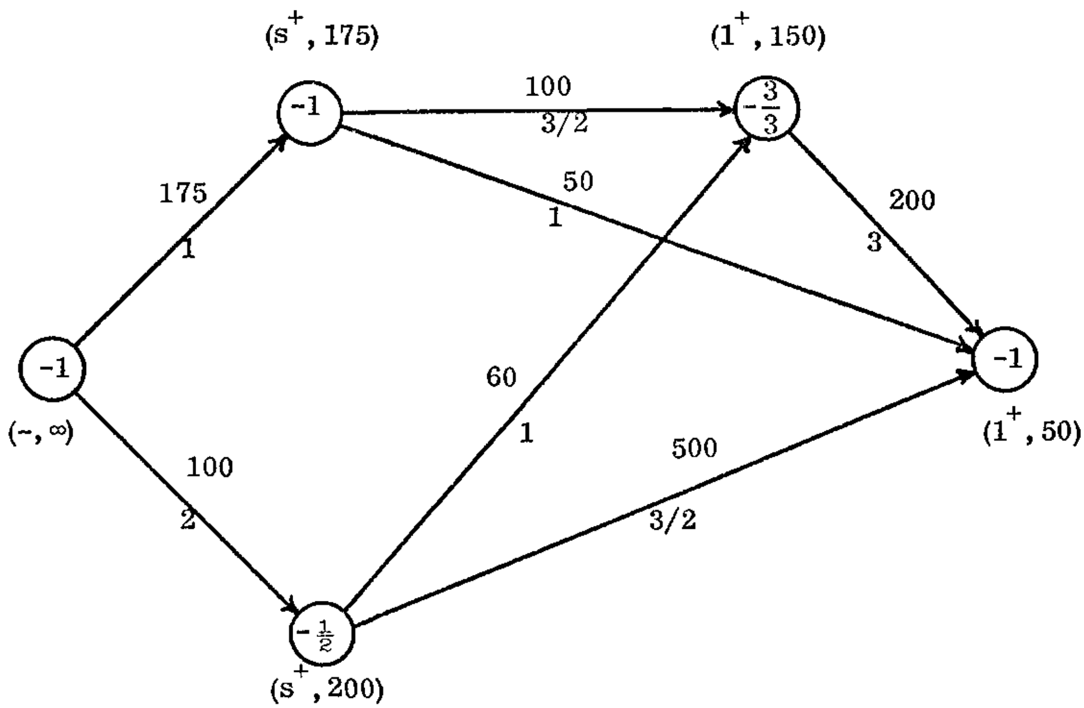

Figure 6. First Iteration.

labels for the second iteration are: $L(s) = (-, \infty)$; $L(1) = (s^+, 125)$; $L(2) = (s^+, 200)$; $L(3) = (1^+, 150)$. Breakthrough is not attained and the set $(X, \bar{X})$ is: $\{(3, t), (1, t), (2, t)\}$.

Step 3 (Dual Variable Change)

The set $A_1 = \{(3, t), (2, t)\}$; $A_2 = \Phi$ and $\theta = \max\{-\infty, -7/9, -2/3; -1\} = -2/3$. Performing the dual variable change we find $\pi'(t) = -1/3$.

Returning to Step 2 and continuing the algorithm we find the optimal solution after six iterations. Table 1 shows the solution.

Consider in the same network, maximize v_t. The optimal solution is obtained in seven iterations and is shown in Table 2.

A final point to make with respect to the algorithm developed related to $k(x, y) = 1, \forall (x, y)$. Initiating Step 1 we find all $\pi(x) = -1$ and all (x, y) with $\pi(x)$

Table 1. Solution for v_s

f(x, y)	v_s	v_t
f(s, 1) = 150	250	800
f(s, 2) = 100		
f(1, 3) = 100		
f(2, 3) = 0		
f(1, t) = 50		
f(2, t) = 200		
f(3, t) = 150		

Table 2. Solution for v_t

f(x, y)	v_s	v_t
f(s, 1) = $143\frac{1}{3}$	$243\frac{1}{3}$	860
f(s, 2) = 100		
f(1, 3) = $93\frac{1}{3}$		
f(2, 3) = 60		
f(1, t) = 50		
f(2, t) = 140		
f(3, t) = 200		

$-k(x, y)\ \pi(y) = 0$. At the first dual variable change $A_1 = A_2 = \Phi$ and $\theta = -1$. Thus when $k(x, y) = 1, \forall (x, y)$ the algorithm degenerates to the maximal flow algorithm described by Ford and Fulkerson (8).

CHAPTER IV

CONCLUSIONS AND RECOMMENDATIONS

The principal results of this thesis are:

a. An efficient algorithm for maximizing flow into an acyclic network with positive gains has been developed and justified.

b. An extension has been shown to use similar techniques to maximize the flow out of the network, i.e. maximize v_t.

Further research is recommended in two general areas:

a. An investigation to extend the algorithm described in this research to consider costs. Considering arc costs results in the following problem formulation:

$$\text{(a)} \quad \text{Maximize } pv_s - \sum_{x,y} a(x,y)\, f(x,y) \tag{4.1}$$

Subject to:

$$\text{(b)} \quad \sum_{y} \left[f(x,y) - k(y,x)\, f(y,x) \right] = \begin{cases} v_s & x = s \\ 0 & x \neq s, t \\ -v_t & x = t \end{cases}$$

$$\text{(c)} \quad 0 \leq f(x,y) \leq c(x,y)$$

$$\text{(d)} \quad v_s,\ v_t \geq 0$$

where $p >> 0$.

The dual of 4.1 is:

$$\text{(a)} \quad \text{Minimize } \sum_{x,y} c(x,y)\, \gamma(x,y) \tag{4.2}$$

(b) $\pi(x) - k(x,y)\,\pi(y) + \gamma(x,z) \geq -\,a(x,y)$

(c) $\pi(s) \leq -p$

(d) $\pi(t) \geq 0$

(e) $\gamma(x,y) \geq 0$

Complementary slackness conditions for 4.1 and 4.2 are:

(1) $\pi(x) - k(x,y)\,\pi(y) > a(x,y) \Rightarrow f(x,y) = 0$

(2) $\pi(x) - k(x,y)\,\pi(y) < a(x,y) \Rightarrow f(x,y) = c(x,y)$

(3) $v_s > 0 \Rightarrow \pi(s) = -p$

(4) $v_t > 0 \Rightarrow \pi(t) = 0$

By choosing p large enough we can start with all $\pi(x) < 0$. The problem arises when we attempt a dual variable change. Specifically consider some arc $(x,y) \epsilon (\bar{X}, \bar{X})$ with $\pi(x) - k(x,y)\,\pi(y) = a(x,y)$ and $0 < f(x,y) < c(x,y)$. In order for the new set of dual variables to maintain complementary slackness conditions for this arc, we must have $\theta = 0$. This would destroy the proof of finiteness. Some other dual variable change in $\bar{X}$ besides $\pi'(x) = (1+\theta)\,\pi(x)$ is needed.

b. Using v_s and/or v_t as given quantities, and considering arc costs consider problems associated with the determination of the optimal set of multipliers.

BIBLIOGRAPHY

1. Balas, Egon. "The Dual Method for the Generalized Distribution Problem," Management Science, Vol 12, No. 7, 1966, pp. 555-568.

2. Balas, E. and P. Ivanescue. "On the Generalized Transportation Problem," Management Science, Vol 11, No. 1, 1964, pp. 188-203.

3. Berge, Claude. The Theory of Graphs and its Applications, John Wiley and Sons, New York, 1966.

4. Berge, Claude and A. Ghouila-Houri. Programming, Games and Transportation Networks, John Wiley and Sons, New York, 1965.

5. Charnes, A. and W. Raike. "One Pass Algorithm for Some Generalized Network Problems," Operations Research, Vol 14, No. 5, 1966, pp. 914-924.

6. Dantzig, G. B. Linear Programming and Extensions, Princeton University Press, Princeton, New Jersey, 1963.

7. Eisemann, K. "The Generalized Stepping Stone Method for the Machine Loading Model," Management Science, Vol 11, No. 1, 1964, pp. 154-177.

8. Ford, L. R., Jr. and D. R. Fulkerson. Flows in Networks, Princeton University Press, Princeton, New Jersey, 1962.

9. Hinkle, R. H. and A. M. Jezior, "The Delivery of Health Care: A Network Flow Problem," talk presented at ORSA Meeting 17 June 1969, Denver, Colorado.

10. Jewell, William S. "Optimal Flow through Networks with Gains," Operations Research, Vol 10, July-August 1962, pp. 476-499.

11. Johnson, E. L., "Networks and Basic Solutions," Operations Research, Vol 14, No. 4, 1966, pp. 619-623.

12. Johnson, E. L., Programming in Networks and Graphs, Operations Research Center, University of California, Berkeley, 1965.

13. Liu, C. L. Introduction to Combinational Mathematics, McGraw Hill Book Company, New York, 1968.

14. Markowitz, H. M. "Concepts and Computing Procedures for Certain X_{ij} Programming Problems," Proceedings of the Second Symposium on in Linear Programming 1 and 2, Directorate of Management Analysis DCS/ Comptroller Headquarters U. S. Air Force, Washington, D. C.

15. Murray, Ralph T. "The Problem of Entry into the Health Care System," Hospital Administration, Vol 13, No. 2, May 1969.

16. Ore, Oystein. Graphs and their Uses, Random House, New York, 1963.

17. Whitehead, C. T. Cities in Trouble: An Agenda for Research, Pascal Anthony H., ed. RM 5603-RC, Rand Corporation, Santa Monica, California, August 1968.